BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:	
Sashikanth Chandrasekaran, et al.	Group Art Unit: 2157
Serial No.: 09/872,891	Examiner: Gold, Avi M.
Filed: May 31, 2001	Confirmation No.: 3158
For: METHOD AND MECHANISM FOR PREDICTING DATA CONFLICTS AND GENERATING A LOAD DISTRIBUTION PLAN IN A MULTI-NODE SYSTEM	

AMENDED APPEAL BRIEF UNDER 37 CFR § 41.37

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Applicant submits this Amended Appeal Brief pursuant to the Notice of Non-Compliant Amendment mailed January 25, 2007. This Amended Appeal Brief is submitted in triplicate and organized in accordance with 37 C.F.R. §41.37(c)(1):

I. REAL PARTY IN INTEREST

The real party in interest is the assignee Oracle Corporation.

II. RELATED APPEALS AND INTERFERENCES

An appeal was filed in May 19, 2006 and an amended appeal was filed on August 28, 2006 in the present case with serial number 09/872,891. The present case was reopened as a result of the appeal and the Office Action dated November 30, 2006 indicating that the case was reopened is included in the *RELATED PROCEEDINGS APPENDIX.*

III. STATUS OF CLAIMS

Claims 1- 66 are pending. Claims 1- 66 are rejected, and are appealed. Claims 1, 14, 32-36, 40 and 54 are independent claims.

IV. STATUS OF AMENDMENTS

Amendment After Final under 37 C.F.R. § 1.116 was filed on September 4, 2007 and no claims were amended in the Amendment After Final.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The present section of the Appeal Brief is set forth merely to comply with the requirements of 37 C.F.R. § 41.37(c) (v) and is not intended to limit the pending claims in any way.

Claim 1 recites:

A method for predicting the behavior of a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) outputting the prediction.

Claim 14 recites:

A method for distributing a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) outputting the workload distribution scheme.

Claim 32 recites:

A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for optimizing the distribution of a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes; and

e) outputting the optimized distribution scheme.

Claim 33 recites:

A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for distributing a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) outputting the workload distribution scheme.

Claim 34 recites:

A system for distributing a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for, based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) means for outputting the workload distribution scheme.

Claim 35 recites:

A system for optimizing the distribution of a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for optimizing the distribution of the workload across the plurality of nodes based on a result of the tracing; and

e) means for outputting the optimized distribution scheme.

Claim 36 recites:

A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for predicting the behavior of a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) outputting the prediction.

Claim 40 recites:

A system for predicting the behavior of a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for, based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) means for outputting the prediction.

Claim 54 recites:

A method for optimizing the distribution of a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes; and

e) outputting the optimized distribution scheme.

An example of a method that includes "executing the workload on a single node before the workload is sent to a plurality of nodes for execution," and "tracing the execution of the workload to identify a potential data conflict," is described in paragraphs 20-26 and figure 3 (see reference numbers 302, 304) of the specification. An example of a method that includes "outputting optimized distribution scheme" is described in paragraphs 31-35, and figure 4 of the specification.

A method that includes "executing the workload on a single node before the workload is sent to a plurality of nodes for execution" allows the execution of the workload in a multi-node environment to be simulated on the single node. This allows for predicting a behavior of the execution of the workload in the multi-node environment before the workload is sent to the multi-node environment for actual execution. For example, if a result of the simulation indicates that it is desirable to execute the workload in a multi-node environment, then the workload may be sent to the multi-node environment for execution by the multi-nodes. On the other hand, if a result of the simulation indicates that it is not desirable to execute the workload in a multi-node environment, then the workload may be executed by the single node, and may not be sent to the multi-node environment for execution.

In particular, with respect to claims 1, 36, and 40, the specification describes embodiments of a method, a computer product, and a system, for: a) receiving a workload to be executed (paragraph 16), b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (paragraphs 20-28 and Figure 3), c) tracing the execution of the workload to identify a potential data conflict (paragraphs 20-28 and Figure 3), d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes (paragraph 14, 20-28 and Figure 3), and e)outputting the prediction (paragraph 26 and Figure 3).

With respect to claims 14, 33, and 34, the specification describes embodiments of a method, a computer product, and a system , for: a) receiving a workload to be executed (paragraph 16), b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (paragraphs 20-26, 32-42, and Figure 3-4), c) tracing the execution of the workload to identify a potential data conflict (paragraphs 20-28 and Figure 3), d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes (paragraphs 32-42, and Figure 4), and e) outputting the workload distribution scheme (paragraphs 32-42 and Figure 4),

With respect to claims 32, 35, and 54, the specification describes embodiments of a method, a computer product, and a system , for: a) receiving a workload to be executed (paragraph 16), b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (paragraphs 20-26, 32-42, and Figure 3-4), c) tracing the execution of the workload to identify a potential data conflict (paragraphs 20-28 and Figure 3), d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes (paragraphs 32-42, and Figure 4), and e) outputting the optimized distribution scheme (paragraphs 31-35, and Figure 4).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The grounds of rejection to be reviewed on appeal are follow: Claims 1-9, 11, 14-58, 60-63, 65, and 66 are allegedly anticipated by U.S. Patent No. 5,937,165 issued to Schwaller et al. (Schwaller) under 35 U.S.C. § 102(b). Claims 10 and 12 stand rejected under 35 U.S.C. 103 as being unpatentable over Schwaller in view of U.S. Patent No. 6,154,813 issued to Martin et al. (Martin). Claims 13, 59, and 64 stand rejected under 35 U.S.C. 103 as being unpatentable over Schwaller in view of U.S. Patent No. 6, 542, 930 issued to Auvenshire et al.

VII. ARGUMENTS

A. Claim Rejections under 35 U.S.C. § 102

Claim 1-9, 11, 14-58, 60-63, 65 and 66 stand rejected under 35 U.S.C. § 102(b) as anticipated by U.S. Patent No. 5,937,165 issued to Schwaller et al. (Schwaller).

Claims 1-9, 11, 14-58, 60-63, 65, and 66

For claim 1, there are one or more claimed limitations that are not disclosed, taught or suggested by the cited references. Claim 1 recites the following limitations:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) outputting the prediction.

1. Claims 14, 32-36, 40, and 54 also recite "tracing the execution of the workload to identify a **potential data conflict**." According to the Advisory Action, Fig. 5A and column 9, lines 46-column 10, line 39, and col. 3, lines 33-48 allegedly disclose the above limitation. Applicants respectfully submit that Schwaller does not disclose or suggest this limitation.

Schwaller is directed toward testing communication networks to obtain **timing measurements** in order to analyze network performance (Abstract and col. 9, line 35). The cited passage in Schwaller discloses test scripts that consist of commands to send and receive data, and generate **timing records** to capture the performance fluctuations with each transaction (table 2, col. 9 lines 46-50, col. 10, line 25, and col. 10, line 39-46). The focus of Schwaller is on the performance of a network in regards to the time that it takes for transactions to execute between nodes. There is nothing in the cited passage of Schwaller that discloses or suggests potential data conflict, much less, identifying a potential data conflict by tracing an execution of workload.

Also according to the Advisory Action, even if Schwaller scripts only disclose monitoring a network to determine the duration of time for nodes to execute transactions, this would allegedly still identify potential data conflicts in the timing for transactions. Applicants respectfully disagree. Determining a potential data conflict is not an inherent step in obtaining a time measurement. Schwaller makes neither mention of identifying data conflicts nor the identification of a data conflict from a time measurement. Schwaller teaches monitoring a network to determine the duration of time for nodes to execute transactions and provides performance characteristics, such as throughput, transaction rate and response time for the test scenario (col. 4, lines 15-20). Instead of identifying data conflicts, Schwaller provides two variations of scripts to obtain time measurements, neither of which involves determining a data conflict. Specifically, Schwaller tests the performance of the network with scripts containing either short or long connections in an effort to vary the impact of the start-up/takedown overhead on the duration of execution of the test script (col. 9 lines 50 through col. 10 line 21). Thus, Schwaller does not disclose or suggest determining a potential data conflict, much less, determining a potential data conflict by tracing an execution of workload.

For the above reasons, Applicant respectfully submits that claims 1, 14, 32-36, 40, and 54, and their respective dependent claims, are patentable over Schwaller under 35 U.S.C. § 102.

B. Claim Rejections under 35 U.S.C. § 103

Claims 10 and 12 stand rejected under 35 U.S.C. 103 as being unpatentable over Schwaller in view of U.S. Patent No. 6,154,813 issued to Martin et al. (Martin).

Claims 10 and 12

1. Applicant submits that claims 10 and 12 are patentable over Schwaller in view of Martin. As discussed, Schwaller does not disclose or suggest the limitations in claim 9 from which claims 10 and 12 depend. Applicants assert that Martin fails to remedy the deficiencies present in Schwaller.

Martin is directed toward a cache management system for buffering media files being simultaneously accessed by multiple clients (Abstract). Martin teaches a cache management strategy

for replacement of data in a cache for a continuous media server, and does not require or teach the execution of a workload on a node to accomplish the cache management strategy. Thus, Martin does not disclose execution of the workload to identify a potential data conflict nor executing the workload on a single node before the workload is sent to a plurality of nodes for execution.

For at least the foregoing reasons, Applicant respectfully submits that claims 10 and 12 are patentable over Schwaller in view of Martin under 35 U.S.C. § 103.

Claims 13, 59, and 64

Claims 13, 59, and 64 stand rejected under 35 U.S.C. 103 as being unpatentable over Schwaller in view of U.S. Patent No. 6, 542, 930 issued to Auvenshire et al. (Auvenshire).

1. Applicant submits that claims 13, 59, and 64 are patentable over Schwaller in view of Auvenshire. As discussed, Schwaller does not disclose or suggest the limitations in claims 1, 54, and 32 from which claims 13, 59, and 64 depend. Applicants assert that Auvenshire fails to remedy the deficiencies present in Schwaller.

Auvenshire is directed toward a distributed file system to improve file distribution, user access, server workload and the like (Abstract). Auvenshire does not disclose execution of the workload to identify a potential data conflict nor executing the workload on a single node before the workload is sent to a plurality of nodes for execution.

For at least the foregoing reasons, Applicant respectfully submits that claims 10 and 12 are patentable over Schwaller in view of Auvenshire under 35 U.S.C. § 103.

VIII. CONCLUSION

For the above reasons, Applicant respectfully submits that rejection of claims 1-66 has been overcome. Accordingly, Applicant requests that the Board of Patent Appeals and Interferences overrule the Examiner and allow claims 1-66.

The Commissioner is authorized to charge any fees due or credit any overpayment in connection with the filing of this document to Bingham McCutchen's Deposit Account No. **50-4047**, referencing billing number **7011112001**. If the Examiner has any questions or comments, please contact the undersigned at the number listed below.

Respectfully submitted,

Dated: February 25, 2008

By: _____
Gerald Chan
Reg. No. 51,541

BINGHAM MCCUTCHEN LLP
Three Embarcadero Center
San Francisco, CA 94111-4067
Telephone: (650) 849-4960
Telefax: (650) 849-4800

APPENDIX A: Pending Claims

Listing of Appealed Claims 1-66.

1. (Previously Presented) A method for predicting the behavior of a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) outputting the prediction.

2. (Original) The method of claim 1 wherein the action of identifying potential data conflicts comprises predicting how many data conflicts will occur.

3. (Original) The method of claim 1 wherein the action of identifying potential data conflicts comprise predicting types of data conflicts.

4. (Original) The method of claim 3 in which the types of data conflicts comprises a read-write conflict.

5. (Original) The method of claim 3 in which the types of data conflicts are based upon types of operations needed to resolve the data conflicts.

6. (Original) The method of claim 3 in which the different types of data conflicts have differing levels of expense associated with operations needed for data conflict resolution.

7. (Original) The method of claim 1 in which the potential data conflicts are at the granularity of a data block.

8. (Original) The method of claim 1 in which the potential data conflicts are identified based upon workload division between sessions.

9. (Original) The method of claim 1 further comprising:

f) selecting a number of nodes;

g) dividing the traced execution of the workload across the number of nodes.

10. (Original) The method of claim 9 in which modulo division is used to divide the traced execution of the workload across the number of nodes.

11. (Original) The method of claim 9 in which the number of nodes corresponds to an anticipated number of nodes for a distributed computing system.

12. (Original) The method of claim 9 in which a modulo class represents a node in the number of nodes.

13. (Original) The method of claim 1 in which the potential data conflicts are used to compute costs of migrating the workload to a distributed system.

14. (Previously Presented) A method for distributing a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) outputting the workload distribution scheme.

15. (Previously Presented) The method of claim 14, wherein the forming the workload distribution scheme comprises determining workload distribution in manner which reduces the potential data conflicts.

16. (Original) The method of claim 14, wherein the workload distribution scheme is based upon data accesses.

17. (Original) The method of claim 16 in which the workload is grouped in the workload distribution scheme to maximize intersection of data access on a same group of nodes.

18. (Original) The method of claim 16 in which the workload is grouped in the workload distribution scheme to minimize intersection of data access across different groups of nodes.

19. (Original) The method of claim 14, wherein the workload distribution scheme is based upon access frequencies.

20. (Original) The method of claim 19 in which data objects accessed by the workload are associated with weighting factors.

21. (Original) The method of claim 20 in which not all the data objects are associated with same weighting factors.

22. (Original) The method of claim 20 in which a weighted correlation is performed between the data objects and entities that access the data objects.

23. (Original) The method of claim 22 in which the entities that access the data objects comprises sessions.

24. (Original) The method of claim 22 in which subsets of the entities that access the data objects are grouped together.

25. (Original) The method of claim 24 in which a data structure is employed to represent an affinity between one of the entities that access the data objects and another of the entities.

26. (Original) The method of claim 14 in which the workload comprises data access upon one or more hierarchical objects.

27. (Original) The method of claim 26 in which tracing the execution of the workload comprises tracing identifiers for the one or more hierarchical objects.

28. (Original) The method of claim 14 in which tracing the execution of the workload comprises tracing identifiers associated with entities that access data.

29. (Original) The method of claim 28 in which the entities comprise sessions.

30. (Original) The method of claim 28 in which the workload distribution scheme distributes the workload based upon partitioning of the entities that access data.

31. (Previously Presented) The method of claim 30 in which an association is formed between partitioning of the entities that access data and partitioning of one or more applications within the workload.

32. (Previously Presented) A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for optimizing the distribution of a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes; and

e) outputting the optimized distribution scheme.

33. (Previously Presented) A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for distributing a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) outputting the workload distribution scheme.

34. (Previously Presented) A system for distributing a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for, based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes; and

e) means for outputting the workload distribution scheme.

35. (Previously Presented) A system for optimizing the distribution of a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for optimizing the distribution of the workload across the plurality of nodes based on a result of the tracing; and

e) means for outputting the optimized distribution scheme.

36. (Previously Presented) A computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for predicting the behavior of a workload across a plurality of nodes, the process comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) outputting the prediction.

37. (Previously Presented) The computer program product of claim 36 wherein the action of identifying potential data conflicts comprises predicting how many data conflicts will occur.

38. (Previously Presented) The computer program product of claim 36 wherein the action of identifying potential data conflicts comprises predicting types of data conflicts.

39. (Previously Presented) The computer program product of claim 36 in which the potential data conflicts are identified based upon workload division between sessions.

40. (Previously Presented) A system for predicting the behavior of a workload across a plurality of nodes, comprising:

a) means for receiving a workload to be executed;

b) means for executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) means for tracing the execution of the workload to identify a potential data conflict;

d) means for, based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes; and

e) means for outputting the prediction.

41. (Previously Presented) The system of claim 40 wherein the means for tracing is configured to predict how many data conflicts will occur.

42. (Previously Presented) The system of claim 40 wherein the means for tracing is configured to predict types of data conflicts.

43. (Previously Presented) The system of claim 40 in which the means for tracing is configured to identify the potential data conflicts based upon workload division between sessions.

44. (Previously Presented) The computer program product of claim 33, wherein the forming the workload distribution scheme comprises determining workload distribution in manner which reduces the potential data conflicts.

45. (Previously Presented) The computer program product of claim 33, wherein the workload distribution scheme is based upon data accesses.

46. (Previously Presented) The computer program product of claim 33, wherein the workload distribution scheme is based upon access frequencies.

47. (Previously Presented) The computer program product of claim 33 in which the workload comprises data access upon one or more hierarchical objects.

48. (Previously Presented) The computer program product of claim 33 in which tracing the execution of the workload comprises tracing identifiers associated with entities that access data.

49. (Previously Presented) The system of claim 34, wherein the means for forming the workload distribution scheme comprises means for determining workload distribution in manner which reduces the potential data conflicts.

50. (Previously Presented) The system of claim 34, wherein the workload distribution scheme is based upon data accesses.

51. (Previously Presented) The system of claim 34, wherein the workload distribution scheme is based upon access frequencies.

52. (Previously Presented) The system of claim 34 in which the workload comprises data access upon one or more hierarchical objects.

53. (Previously Presented) The system of claim 34 in which the means for tracing the execution of the workload comprises means for tracing identifiers associated with entities that access data.

54. (Previously Presented) A method for optimizing the distribution of a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed;

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution;

c) tracing the execution of the workload to identify a potential data conflict;

d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes; and

e) outputting the optimized distribution scheme.

55. (Previously Presented) The method of claim 54, wherein the action of identifying potential data conflicts comprises predicting how many data conflicts will occur.

56. (Previously Presented) The method of claim 54, wherein the action of identifying potential data conflicts comprise predicting types of data conflicts.

57. (Previously Presented) The method of claim 54 in which the potential data conflicts are at the granularity of a data block.

58. (Previously Presented) The method of claim 54 in which the potential data conflicts are identified based upon workload division between sessions.

59. (Previously Presented) The method of claim 54 in which the potential data conflicts are used to compute costs of migrating the workload to a distributed system.

60. (Previously Presented) The computer program product of claim 32, wherein the action of identifying potential data conflicts comprises predicting how many data conflicts will occur.

61. (Previously Presented) The computer program product of claim 32, wherein the action of identifying potential data conflicts comprise predicting types of data conflicts.

62. (Previously Presented) The computer program product of claim 32 in which the potential data conflicts are at the granularity of a data block.

63. (Previously Presented) The computer program product of claim 32 in which the potential data conflicts are identified based upon workload division between sessions.

64. (Previously Presented) The computer program product of claim 32 in which the potential data conflicts are used to compute costs of migrating the workload to a distributed system.

65. (Previously Presented) The system of claim 35, wherein the means for tracing is configured to predict how many data conflicts will occur.

66. (Previously Presented) The system of claim 35, wherein the means for tracing is configured to predict types of data conflicts.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

See Attached.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/872,891	05/31/2001	Sashikanth Chandrasekaran	256/145	3158

23639 7590 11/30/2006

BINGHAM, MCCUTCHEN LLP
THREE EMBARCADERO CENTER
18 FLOOR
SAN FRANCISCO, CA 94111-4067

EXAMINER	
GOLD, AVI M	

ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 11/30/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

PCM

Docket: OI 701111 - 2001
Action: Resp Due
Date Due: 2/28/2007
FD is 5/28/2007
CAL
KM

Office Action Summary	Application No.	Applicant(s)	
	09/872,891	CHANDRASEKARAN ET AL	
	Examiner	Art Unit	
	Avi Gold	2157	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *01 September 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-66* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-66* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1 ☐ Certified copies of the priority documents have been received.

2 ☐ Certified copies of the priority documents have been received in Application No. _____.

3 ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

This action is responsive to the appeal filed on September 1, 2006. Claims 1-66 are pending.

Response to Amendment

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 1-9, 11, 14-58, 60-63, 65, and 66 are rejected under 35 U.S.C. 102(b) as being anticipated by Schwaller et al., U.S. Patent No. 5,937,165.

Schwaller teaches the invention substantially as claimed including systems, methods, and computer program products for performance testing of computer networks (see abstract).

Regarding claims 1, 36, and 40, Schwaller teaches a method, computer program product, and system for predicting the behavior of a workload across a plurality of nodes, comprising:

a) receiving a workload to be executed (col. 9, lines 24-27, Schwaller discloses a script provided);

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (fig. 5, col. 9, lines 24-35, Schwaller discloses a script provided to a single endpoint node to be tested on that node);

c) tracing the execution of the workload to identify a potential data conflict (fig. 5A, table 2, col. 9, line 46 – col. 10, line 39, Schwaller discloses that the test is monitored to identify conflicts);

d) based on a result of the tracing, predicting the behavior of the workload across the plurality of nodes (fig. 5A, col. 3, lines 8-11, col. 25, lines 58-64, Schwaller discloses analyzing the performance and making a prediction with performance measurements); and

e) outputting the prediction (col. 25, lines 58-64).

Regarding claims 2, 37, 41, 55, 60, and 65, Schwaller teaches the method, computer program product, and system of claims 1, 36, 40, 54, 32, and 35 wherein the action of identifying potential data conflicts comprises predicting how many data conflicts will occur (col. 9, line 46 – col. 10, line 39).

Regarding claims 3, 38, 42, 56, 61, and 66, Schwaller teaches the method, computer program product, and system of claims 1, 36, 40, 54, 32, and 35 wherein the action of identifying potential data conflicts comprise predicting types of data conflicts (col. 9, line 46 – col. 10, line 39).

Regarding claim 4, Schwaller teaches the method of claim 3 in which the types of data conflicts comprises a read-write conflict (table 2).

Regarding claim 5, Schwaller teaches the method of claim 3 in which the types of data conflicts are based upon types of operations needed to resolve the data conflicts (table 2).

Regarding claim 6, Schwaller teaches the method of claim 3 in which the different types of data conflicts have differing levels of expense associated with operations needed for data conflict resolution (table 2).

Regarding claims 7, 57, and 62, Schwaller teaches the method and computer program product of claims 1, 54, and 32 in which the potential data conflicts are at the granularity of a data block (table 2).

Regarding claims 8, 39, 43, 58, and 63, Schwaller teaches the method, system, and computer program product of claims 1, 36, 40, 54, and 32 in which the potential data conflicts are identified based upon workload division between sessions (table 2).

Regarding claim 9, Schwaller teaches the method of claim 1 further comprising:

f) selecting a number of nodes;

g) dividing the traced execution of the workload across the number of nodes (fig. 5, 5A, Schwaller discloses multiple nodes monitored).

Regarding claim 11, Schwaller teaches the method of claim 9 in which the number of nodes corresponds to an anticipated number of nodes for a distributed computing system (fig. 5, 5A).

Regarding claims 14, 33, 34, and 54, Schwaller teaches a method, computer program product, and system for distributing a workload across a plurality of nodes, the method comprising:

a) receiving a workload to be executed (col. 9, lines 24-37);

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (fig. 5, col. 9, lines 24-35);

c) tracing the execution of the workload to identify a potential data conflict (fig. 5A, table 2, col. 9, line 46 – col. 10, line 39);

d) based on a result of the tracing, forming a workload distribution scheme that distributes the workload across the plurality of nodes (fig. 5A, col. 3, lines 8-11, col. 25, lines 58-64); and

e) outputting the workload distribution scheme (col. 25, lines 58-64).

Regarding claims 15, 44, and 49, Schwaller teaches the method, computer program product, and system of claims 14, 33, and 34, wherein the forming the workload distribution scheme comprises determining workload distribution in manner which reduces the potential data conflicts (fig. 5A, table 2, col. 9, line 46 – col. 10, line 39).

Regarding claims 16, 45, and 50, Schwaller teaches the method, computer program product, and system of claims 14, 33, and 34, wherein the workload distribution scheme is based upon data accesses (table 2).

Regarding claim 17, Schwaller teaches the method of claim 16 in which the workload is grouped in the workload distribution scheme to maximize intersection of data access on a same group of nodes (table 2).

Regarding claim 18, Schwaller teaches the method of claim 16 in which the workload is grouped in the workload distribution scheme to minimize intersection of data access across different groups of nodes (table 2).

Regarding claims 19, 46, and 51, Schwaller teaches the method, computer program product, and system of claims 14, 33, and 34, wherein the workload distribution scheme is based upon access frequencies (table 2).

Regarding claim 20, Schwaller teaches the method of claim 19 in which data objects accessed by the workload are associated with weighting factors (table 2).

Regarding claim 21, Schwaller teaches the method of claim 20 in which not all the data objects are associated with same weighting factors (table 2).

Regarding claim 22, Schwaller teaches the method of claim 20 in which a weighted correlation is performed between the data objects and entities that access the data objects (table 2).

Regarding claim 23, Schwaller teaches the method of claim 22 in which the entities that access the data objects comprises sessions (table 2).

Regarding claim 24, Schwaller teaches the method of claim 22 in which subsets of the entities that access the data objects are grouped together (table 2).

Regarding claim 25, Schwaller teaches the method of claim 24 in which a data structure is employed to represent an affinity between one of the entities that access the data objects and another of the entities (table 2).

Regarding claims 26, 47, and 52, Schwaller teaches the method, computer program product, and system of claims 14, 33, and 34 in which the workload comprises data access upon one or more hierarchical objects (figs. 2-5).

Regarding claim 27, Schwaller teaches the method of claim 26 in which tracing the execution of the workload comprises tracing identifiers for the one or more hierarchical objects (figs. 2-5).

Regarding claims 28, 48, and 53, Schwaller teaches the method, computer program product, and system of claims 14, 33, and 34 in which tracing the execution of the workload comprises tracing identifiers associated with entities that access data (table 2).

Regarding claim 29, Schwaller teaches the method of claim 28 in which the entities comprise sessions (table 2).

Regarding claim 30, Schwaller teaches the method of claim 28 in which the workload distribution scheme distributes the workload based upon partitioning of the entities that access data (table 2).

Regarding claim 31, Schwaller teaches the method of claim 30 in which an association is formed between partitioning of the entities that access data and partitioning of one or more applications within the workload (table 2).

Regarding claims 32 and 35, Schwaller teaches a computer program product that includes a medium usable by a processor, the medium comprising a sequence of instructions which, when executed by said processor, causes said processor to execute a process for optimizing the distribution of a workload across a plurality of nodes, the process and system comprising:

a) receiving a workload to be executed (col. 9, lines 24-27);

b) executing the workload on a single node before the workload is sent to a plurality of nodes for execution (fig. 5, col. 9, lines 24-35);

c) tracing the execution of the workload to identify a potential data conflict (fig. 5A, table 2, col. 9, line 46 – col. 10, line 39);

d) based on a result of the tracing, optimizing the distribution of the workload across the plurality of nodes (fig. 5A, col. 3, lines 8-11, col. 25, lines 58-64); and

e) outputting the optimized distribution scheme (col. 25, lines 58-64).

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> **(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.**

4. Claims 10 and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schwaller et al., U.S. Patent No. 5,937,165, further in view of Martin et al., U.S. Patent No. 6,154,813.

Schwaller teaches the invention substantially as claimed including systems, methods, and computer program products for performance testing of computer networks (see abstract).

As to claims 10 and 12, Schwaller teaches the method of claim 9.

Schwaller fails to teach the limitation further including the use of modulo division to divide the traced execution of the workload across the number of nodes and the use of a modulo class to represent a node in the number of nodes.

However, Martin teaches a cache management scheme for continuous media data, such as audio or video (see abstract). Martin teaches the use of modulo division (col. 4, lines 1-15, col. 5, lines 46-63).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Schwaller in view of Martin to use modulo division and a modulo class in association with nodes. One would be motivated to do so because they are efficient ways of organizing nodes.

5. Claims 13, 59, and 64 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schwaller et al., U.S. Patent No. 5,937,165, further in view of Auvenshine, U.S. Patent No. 6,542,930.

Schwaller teaches the invention substantially as claimed including systems, methods, and computer program products for performance testing of computer networks (see abstract).

As to claims 13, 59, and 64, Schwaller teaches the method and computer program product of claims 1, 54, and 32.

Schwaller fails to teach the limitation further including the potential data conflicts being used to compute costs of migrating the workload to a distributed system.

However, Auvenshine teaches a distributed file system with automated file management achieved by decoupling data analysis and movement operations (see abstract). Auvenshine teaches the use of a distributed system.

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Schwaller in view of Auvenshine to migrate the workload to a distributed system. One would be motivated to do so because it would still seem as if the system is one local machine.

Response to Arguments

6. In view of the appeal brief filed on September 1, 2006, PROSECUTION IS HEREBY REOPENED. A new ground of rejection is set forth above.

To avoid abandonment of the application, appellant must exercise one of the following two options:

(1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,

(2) request reinstatement of the appeal.

If reinstatement of the appeal is requested, such request must be accompanied by a supplemental appeal brief, but no new amendments, affidavits (37 CFR 1.130, 1.131 or 1.132) or other evidence are permitted. See 37 CFR 1.193(b)(2).

Conclusion

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Pat. No. 6,427,166 to Hurst et al.

U.S. Pat. No. 5,928,344 to Stierli.

U.S. Pat. No. 6,681,251 to Leymann et al.

U.S. Pat. No. 6,442,564 to Frey et al.

U.S. Pat. No. 5,819,033 to Caccavale.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Avi Gold whose telephone number is 571-272-4002. The examiner can normally be reached on M-F 8:00-5:30 (1st Friday Off).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on 571-272-4001. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Avi Gold

Patent Examiner

Art Unit 2157

AMG

ARIO ETIENNE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Notice of References Cited	Application/Control No. 09/872,891	Applicant(s)/Patent Under Reexamination CHANDRASEKARAN ET AL	
	Examiner Avi Gold	Art Unit 2157	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-6,405,257 B1	06-2002	Gersht et al.	709/235
*	B	US-6,154,813 A	11-2000	Martin et al.	711/133
*	C	US-6,542,930 B1	04-2003	Auvenshine, John Jason	709/224
*	D	US-6,427,166 B1	07-2002	Hurst et al.	709/220
*	E	US-5,928,344 A	07-1999	Stierli, Peter	710/105
*	F	US-6,681,251 B1	01-2004	Leymann et al.	709/226
*	G	US-6,442,564 B1	08-2002	Frey et al.	707/103R
*	H	US-5,819,033 A	10-1998	Caccavale, Frank Samuel	709/224
*	I	US-5,657,450	08-1997	Rao et al.	707/10
*	J	US-5,937,165	08-1999	Schwaller et al.	709/224
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.